课程回顾

■插入排序分析

➢循环不变式、算法执行时间

■渐近表示法

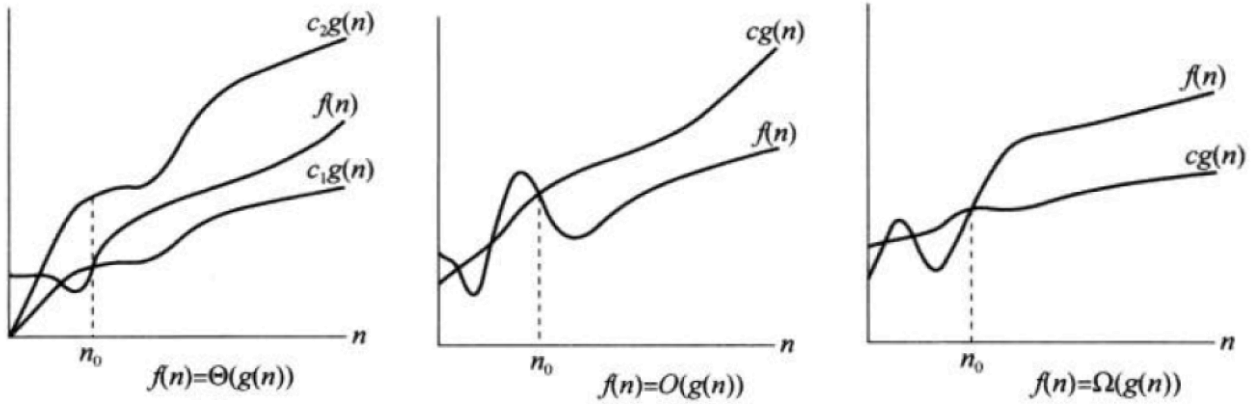
➢Θ记号

➢O记号

➢Ω记号

➢o记号

➢ω记号



$$\lim_{n\to\infty}\frac{f(n)}{g(n)}=\begin{cases}0 & f(n)=o(g(n)),f(n)=O(g(n))\\ a,a>0 & f(n)=\Theta(g(n)),f(n)=O(g(n)),f(n)=\Omega(g(n))\\ \infty & f(n)=\omega(g(n)),f(n)=\Omega(g(n))\end{cases}$$

■函数比较（传递性、自反性、对称性、转置对称性）

函数比较

- 函数比较：实数的许多关系性质也适用于渐近比较。假定$f(n)$和$g(n)$是渐近正的：
- 传递性 (5种渐近记号均适用)

$$
\begin{aligned}
f(n) = \Theta(g(n)) \text{ and } g(n) = \Theta(h(n)) &\Rightarrow f(n) = \Theta(h(n)) \\
f(n) = O(g(n)) \text{ and } g(n) = O(h(n)) &\Rightarrow f(n) = O(h(n)) \\
f(n) = \Omega(g(n)) \text{ and } g(n) = \Omega(h(n)) &\Rightarrow f(n) = \Omega(h(n)) \\
f(n) = o(g(n)) \text{ and } g(n) = o(h(n)) &\Rightarrow f(n) = o(h(n)) \\
f(n) = \omega(g(n)) \text{ and } g(n) = \omega(h(n)) &\Rightarrow f(n) = \omega(h(n))
\end{aligned}
$$

函数比较 (续)

- **自反性** (渐近非紧界无该性质)

$$f(n) = \Theta(f(n)), f(n) = O(f(n)), f(n) = \Omega(f(n))$$

- **对称性** (仅紧致界有该性质)

$$f(n) = \Theta(g(n)) \text{ iff } g(n) = \Theta(f(n))$$

- **转置对称性** (O与Ω、o与ω)

$$f(n) = O(g(n)) \text{ iff } g(n) = \Omega(f(n))$$
$$f(n) = o(g(n)) \text{ iff } g(n) = \omega(f(n))$$

函数比较 (续)

■可将两函数间的渐近比较类比于两个实数间的比较：

$$f(n) = O(g(n)) \text{ is like } a \leq b$$
$$f(n) = \Omega(g(n)) \text{ is like } a \geq b$$
$$f(n) = \Theta(g(n)) \text{ is like } a = b$$
$$f(n) = o(g(n)) \text{ is like } a < b$$
$$f(n) = \omega(g(n)) \text{ is like } a > b$$

- 若$f(n)=o(g(n))$，称$f(n)$渐近小于$g(n)$
- 若$f(n)=\omega(g(n))$，称$f(n)$渐近大于$g(n)$

函数比较 (续)

- 但实数的三岐性（三分性）不可类比到渐近表示中
- 三岐性：对任意两个实数a和b，下列三种情况必有一个成立：$a<b$, $a=b$, $a>b$
- 并非所有函数都是渐近可比较的，即存在$f(n)$和$g(n)$，$f(n)\neq O(g(n))$且$f(n)\neq\Omega(g(n))$
  - 例：函数n和$n^{1+\sin n}$无法渐近比较
    - $1+\sin n$在0到2之间波动

本章内容

- 算法定义及基本概念（教材Chapter 1）
- 算法描述（教材Chapter 2）
- 函数增长及渐近记号表示（教材Chapter 3）
- 标准记号与常用函数（教材Chapter 3）
- NP完全性理论（区分并理解P/ NP/ NPC/ NP-Hard 几类问题）（教材Chapter 34）

标准记号

■单调性

- 若$m \le n \rightarrow f(m) \le f(n)$，则函数$f(n)$是单调递增的
- 若$m \le n \rightarrow f(m) \ge f(n)$，则函数$f(n)$是单调递减的
- 若$m < n \rightarrow f(m) < f(n)$，则函数$f(n)$是严格递增的
- 若$m < n \rightarrow f(m) > f(n)$，则函数$f(n)$是严格递减的

标准记号 (续)

■向下取整与向上取整

- $\lfloor x \rfloor$：小于或等于x的最大整数 (x是任意实数)
- $\lceil x \rceil$：大于或等于x的最小整数 (x是任意实数)

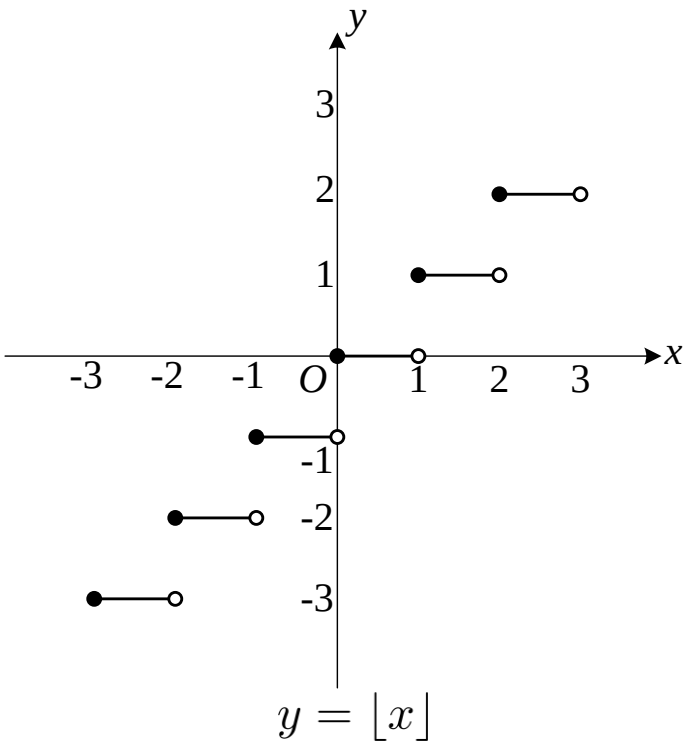
—些性质：

- $x - 1 < \lfloor x \rfloor \leq x \leq \lceil x \rceil < x + 1, \quad \forall x \in \mathbb{R}$
- $\lceil n/2 \rceil + \lfloor n/2 \rfloor = n, \quad \forall n \in \mathbb{Z}$
- $$\left\lceil \frac{\lceil x/a \rceil}{b} \right\rceil = \left\lceil \frac{x}{ab} \right\rceil, \left\lfloor \frac{\lfloor x/a \rfloor}{b} \right\rfloor = \left\lfloor \frac{x}{ab} \right\rfloor,$$
  $$\left\lceil \frac{a}{b} \right\rceil \leq \frac{a + (b-1)}{b}, \left\lfloor \frac{a}{b} \right\rfloor \geq \frac{a - (b-1)}{b}, \quad \forall x \in \mathbb{R} \wedge x \geq 0$$
  $$\forall a, b \in \mathbb{Z} \wedge a, b > 0$$

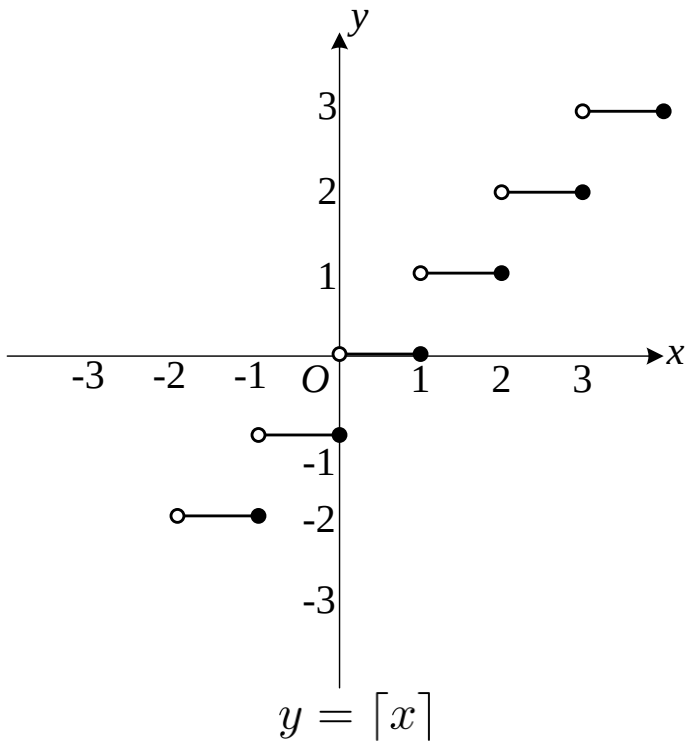
标准记号 (续)

■向下取整与向上取整

- $f(x) = \lfloor x \rfloor$ 和 $f(x) = \lceil x \rceil$ 是单调递增的



$y = \lfloor x \rfloor$



$y = \lceil x \rceil$

标准记号 (续)

■模运算

- 对任意整数a和任意正整数n，$a \bmod n$的值就是商a/n的余数：

$$a \bmod n = a - n\lfloor a/n \rfloor$$

- $0 \leq (a \bmod n) < n$
- 若$(a \bmod n) = (b \bmod n)$，则记$a \equiv b (\bmod n)$，称模$n$时$a$等价于$b$ (表示余数相等)

$$a \equiv b (\bmod n) \quad \text{iff} \quad n\text{是}b\text{-}a\text{的一个因子}$$

- 若模n时a不等价于b，则记$a \not\equiv b (\bmod n)$

常用函数

■多项式

➢n的d次多项式：

$$p(n) = \sum_{i=0}^{d} a_i n^i$$

其中，d是非负整数，常量$a_0, a_1, \ldots, a_d$是多项式的系数且$a_d \neq 0$

常用函数 (续)

■多项式

$$p(n) = \sum_{i=0}^{d} a_i n^i$$

一些性质：

- 一个多项式渐近为正，当且仅当$a_d > 0$
- 对于一个d次渐近正的多项式$p(n)$，有$p(n)=\Theta(n^d)$
- 对任意实常量$a \geq 0$，函数n^a单调递增；对任意实常量$a \leq 0$，函数n^a单调递减
- 若对某个常量k有$f(n)=O(n^k)$，则称函数$f(n)$是多项式有界的

常用函数 (续)

■指数

➢对所有正实数a、实数m和n，有

$$a^0 = 1$$

$$a^1 = a$$

$$a^{-1} = 1/a$$

$$(a^m)^n = a^{mn}$$

$$(a^m)^n = (a^n)^m$$

$$a^m a^n = a^{m+n}$$

对所有n和$a \geq 1$，函数a^n关于n单调递增

为了方便假定$0^0=1$

常用函数 (续)

■指数

—些性质：

➢对所有实常量a和b，其中$a>1$，有

$$\lim_{n\to\infty}\frac{n^b}{a^n}=0$$

因此$n^b=o(a^n)$，即任意底大于1的指数函数比任意多项式函数增长得快

常用函数 (续)

■指数

一些性质：

- 对所有实数x，有

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \ldots = \sum_{i=0}^{\infty} \frac{x^i}{i!}$$

- 对所有实数x，有 $e^x \geq 1+x$，等号当且仅当$x=0$时成立
- 当$|x|<1$时，有 $1+x \leq e^x \leq 1+x+x^2$
- 当$x \to 0$时，$e^x = 1+x+\Theta(x^2)$
- $\lim_{n\to\infty} \left(1 + \frac{x}{n}\right)^n = e^x$

常用函数 (续)

■对数

$$\lg n = \log_2 n$$
$$\ln n = \log_e n$$
$$\lg^k n = (\lg n)^k$$
$$\lg \lg n = \lg(\lg n)$$

- 对数函数仅作用于下一项，例如$\lg n+k=(\lg n)+k$，而不是$\lg(n+k)$

常用函数 (续)

■对数

一些性质：

- 若常量$b>1$，则函数$\log_b n$是严格递增的
- 对所有实数$a>0$, $b>0$, $c>0$, n，有：

$$a = b^{\log_b a}$$

$$\log_b(1/a) = -\log_b a$$

$$\log_c(ab) = \log_c a + \log_c b$$

$$\log_b a = \frac{1}{\log_a b}$$

$$\log_b a^n = n \log_b a$$

$$a^{\log_b c} = c^{\log_b a}$$

$$\log_b a = \frac{\log_c a}{\log_c b}$$

对数的底更换时仅相差一个常量因子

其中对数的底均不为1

常用函数 (续)

证明 $a^{\log_b c} = c^{\log_b a}$

$$a^{\log_b c} = b^{\log_b (a^{\log_b c})} = b^{\log_b c \log_b a} = b^{\log_b (c^{\log_b a})} = c^{\log_b a}$$

常用函数 (续)

■对数

一些性质：

- 当|x|<1时，存在一种简单的级数展开：

$$\ln(1+x) = x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \frac{x^5}{5} - \ldots = \lim_{n\to\infty} \sum_{i=1}^{n} (-1)^i \cdot \frac{x^i}{i}$$

- 若x>-1，则 $\frac{x}{1+x} \le \ln(1+x) \le x$，当且仅当x=0时等号成立

常用函数 (续)

■对数

- 若对某个常量k，$f(n)=O(\lg^k n)$，则称函数$f(n)$是多对数有界的
- $\lim_{n\to\infty}\frac{n^b}{a^n}=0\ (a>1)$ 式中用$\lg n$代替n、2^a代替a，得到：

$$\lim_{n\to\infty}\frac{\lg^k n}{(2^a)^{\lg n}}=\lim_{n\to\infty}\frac{\lg^k n}{n^a}=0\ (a>0)$$

即对任意常量$a>0$，$\lg^k n=o(n^a)$

任意正的多项式函数都比任意多对数函数增长得快

常用函数 (续)

■阶乘

- $n! = 1 \cdot 2 \cdot 3 \cdot \ldots \cdot n = \prod_{i=1}^{n} i$，并定义0!=1
- $$n! = \begin{cases} 1, & n = 0 \\ n \cdot (n-1)!, & n > 0 \end{cases}$$

- 斯特林(Stirling)近似公式

$$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n \left(1 + \Theta\left(\frac{1}{n}\right)\right)$$

常用函数 (续)

■阶乘

➢斯特林(Stirling)近似公式

$$n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n\left(1+\Theta\left(\frac{1}{n}\right)\right)$$

➢对所有$n \geq 1$，有 $n! = \sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\alpha_n}$

其中 $\frac{1}{12n+1} < \alpha_n < \frac{1}{12n}$

➢ $n! = o(n^n)$ $\quad n! = \omega(2^n)$ $\quad \lg(n!) = \Theta(n \lg n)$

常用函数 (续)

■多重函数

- 使用记号$f^{(i)}(n)$表示函数$f(n)$重复i次作用于n上：

$$f^{(i)}(n) = \underbrace{f(f(f...f}_{i}(n)))$$

其中，定义$f^{(0)}(n)=n$

- $$f^{(i)}(n) = \begin{cases} n, & i = 0 \\ f(f^{(i-1)}(n)), & i > 0 \end{cases} \quad (i \in \mathbb{Z})$$

常用函数 (续)

■多重对数函数

- 使用记号$\lg^* n$表示多重对数，定义如下：

$$\lg^* n = \min\{i \geq 0 : \lg^{(i)} n \leq 1\}$$

通俗理解：对于值n，至少要作用几次lg函数才能得到不超过1的结果？

$$n \in (\underbrace{2^{2^{\cdot^{\cdot^{\cdot^2}}}}}_{(\lg^* n)-1}, \underbrace{2^{2^{\cdot^{\cdot^{\cdot^2}}}}}_{\lg^* n}]$$

- 例如：$\lg^*2=1$，$\lg^*4=2$，$\lg^*16=3$，$\lg^*65536=4$，$\lg^*(2^{65536})=5$
- 多重对数是一个增长非常慢的函数
- 可探测的宇宙中原子数目估计约为$10^{80} << 2^{65536}$，因此很少遇到使得$\lg^* n>5$的输入规模n

常用函数 (续)

■斐波那契数

- 每个斐波那契数都是两个前面的数之和：

$$F_0 = 0$$
$$F_1 = 1$$
$$F_i = F_{i-1} + F_{i-2}, \quad i \geq 2$$

- 产生的序列为：0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...

常用函数 (续)

■斐波那契数

➢斐波那契数与黄金分割率ϕ

$$F_i = \frac{\phi^i - \hat{\phi}^i}{\sqrt{5}} = \left\lfloor \frac{\phi^i}{\sqrt{5}} + \frac{1}{2} \right\rfloor$$

斐波那契数以指数形式增长

$\hat{\phi}$是ϕ的共轭数，它们是方程x^2-x-1=0的两个根：

$$\phi = \frac{1+\sqrt{5}}{2} \approx 1.618$$

$$\hat{\phi} = \frac{1-\sqrt{5}}{2} \approx -0.618$$

渐近增长例

■请根据增长的阶从小到大排序以下函数。若$f(n)=o(g(n))$则$f(n)$排在$g(n)$前面；若$f(n)=\Theta(g(n))$则放在同一集合中用大括号括起来。

3	2^n	$5n$	10^{10}	2^{2n}
$n\lg(n+2)$	$\lg\lg n$	n^3	n^n	2^{2^n}
$\lg(n!)$	$n!$			

强调：本课程中约定lg函数表示以2为底的对数（参见教材第32页）

渐近增长例 (续)

3	2^n	$5n$	10^{10}	2^{2n}
$n\lg(n+2)$	$\lg\lg n$	n^3	n^n	2^{2^n}
$\lg(n!)$	$n!$			

■排序结果：

$\{3, 10^{10}\}, \lg\lg n, 5n, \{n\lg(n+2), \lg(n!)\}, n^3, 2^n, 2^{2n}, n!, n^n, 2^{2^n}$

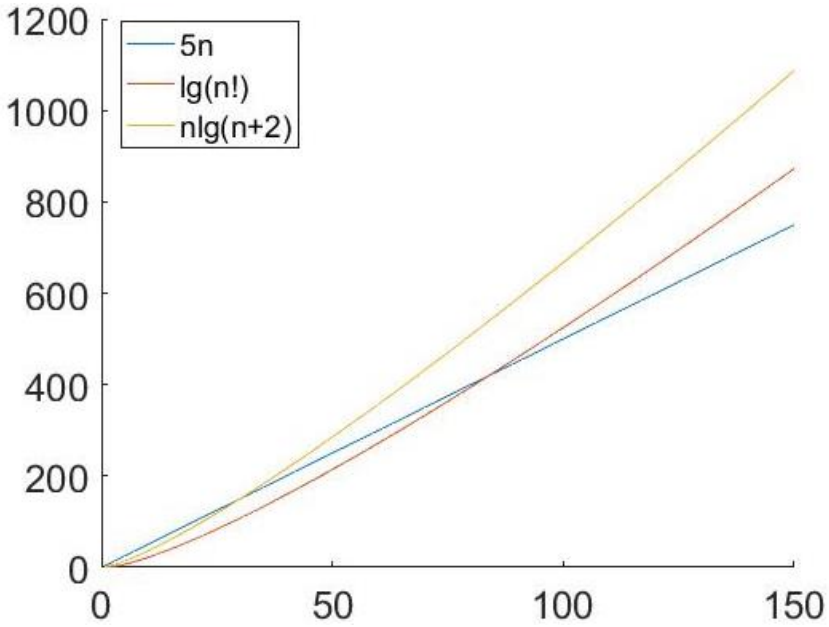
➢采用相除求极限的方法证明排序，可使用洛必达法则

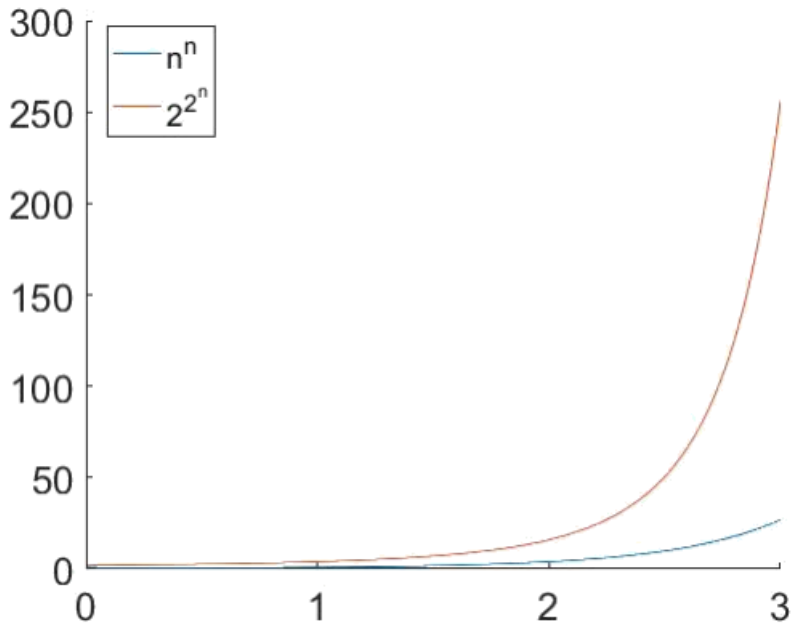
➢例：$\lim\limits_{n\to\infty} \dfrac{\lg\lg n}{5n} = \lim\limits_{n\to\infty} \dfrac{\frac{1}{\lg n\cdot\ln 2}\cdot\frac{1}{n\ln 2}}{5} = \lim\limits_{n\to\infty} \dfrac{1}{5(\ln 2)^2 n\lg n} = 0$

$$\lim_{n\to\infty} \frac{2^n}{2^{2n}} = \lim_{n\to\infty} \frac{2^n}{4^n} = \lim_{n\to\infty} \frac{1}{2^n} = 0$$

$$\lim_{n\to\infty} \frac{n^n}{2^{2^n}} = \lim_{n\to\infty} \frac{2^{n\lg n}}{2^{2^n}} = \lim_{n\to\infty} 2^{n\lg n - 2^n} = 0$$

渐近增长例 (续)


1200
1000
800
600
400
200
0
0
50
100
150
5n
lg(n!)
nlg(n+2)


300
250
200
150
100
50
0
0
1
2
3
n^n
2^{2^n}

本章内容

- 算法定义及基本概念（教材Chapter 1）
- 算法描述（教材Chapter 2）
- 函数增长及渐近记号表示（教材Chapter 3）
- 标准记号与常用函数（教材Chapter 3）
- NP完全性理论（区分并理解P/ NP/ NPC/ NP-Hard 几类问题）（教材Chapter 34）

NP完全性理论

- 从计算的观点来看，要解决的问题的内在复杂性如何？它是“易”计算的还是“难”计算的？
- 知道了一个问题的计算时间下界，就知道了对于该问题能设计出多有效的算法，从而可以较正确地评价对该问题提出的各种算法的效率，进而确定对已有算法还有多少改进的余地
- 许多情况下，要确定一个问题的内在计算复杂性是很困难的

NP完全性理论 (续)

- 多项式时间的算法：对于规模为n的输入，在最坏情况下的运行时间是$O(n^k)$，其中k为某一确定常数

$$O(1) < O(n) < O(n \lg n) < O(n^2) < O(2^n) < O(n!) < O(n^n)$$

多项式级别　　超多项式级别

NP完全性理论 (续)

- 一般来说，我们认为在多项式时间内可解的问题是易处理的问题，在超多项式时间内解决的问题是不易处理的问题
- “NP完全”(NPC) 问题：迄今为止，既没有人找出求解这类问题的多项式时间算法，也没有人能够证明对这类问题不存在多项式时间算法

判定问题与优化问题

- 最优化问题 (optimization problem)：每一个可行解都有一个关联的值，我们希望找出一个具有最佳值的可行解
  - 例如：SHORTEST-PATH问题：已知无向图G以及顶点u和v，要找出u和v之间经过边数目最少的一条路径。在无权、无向图中的单点对间最短路径问题
- 判定问题 (decision problem)：问题的答案是简单的“是”或“否”（或者更形式化地，答案是“1”或“0”）
  - 例如：SHORTEST-PATH相关的判定问题PATH：判定给定的无向图G、顶点u和v、一个整数k，在u和v之间是否存在一条至多包含k条边的路径

判定问题与优化问题 (续)

- NPC类问题只局限于判定问题
- 通常通过对优化的值强加一个界，就可以将一个给定的优化问题转化为一个相关的判定问题
- 试图证明最优化问题“不易处理”时，就可利用该问题与相关的判定问题之间的关系
  - 判定问题要“更容易一些”，至少“不会更难”
  - 如果能够证明某个判定问题是困难问题，相当于证明了其相关最优化问题也是困难的

P类问题

- P类问题 (Polynomial Problem)：在多项式时间内可以解决的问题
  - 例如：排序问题可在$O(n^2)$时间复杂度内解决

NP类问题

- NP类问题 (Non-deterministic Polynomial Problem)：在多项式时间内可以被验证的问题
  - 非确定性算法：猜测 + 验证，猜测阶段是非确定性的，给出问题解的一个猜测；验证阶段是确定性的，验证阶段给出解的正确性
  - 设算法A是解一个判定问题Q的非确定性算法，如果算法A的验证阶段可以在多项式时间内完成，则称算法A是一个多项式时间非确定性算法，也称问题Q是非确定性多项式时间可解的（NP问题）

NP类问题 (续)

- NP类问题 (Non-deterministic Polynomial Problem)：在多项式时间内可以被验证的问题
  - 解可能不好找，但可猜测一个解，很容易验证是否正确
  - $P \subseteq NP$
  - P类问题是否是NP类问题的真子集，目前是一个开放问题

NPC类问题

- 如果一个NP问题和其他任何NP问题一样“不易解决”，那么我们认为这一问题是NPC类问题或称之为NP完全问题
- 如果任何NP完全问题可以在多项式时间内解决，那么所有NP问题都有一个多项式时间算法
- 若能够确定一个NP完全问题，就可以提供充分的论据说明其不易处理性
  - 采用近似算法或解决某种易处理问题的特例，而不是寻找求的问题精确解的快速算法